

Method and Apparatus for processing two or more initially decoded audio signals received or replayed from a bitstream

The invention relates to a method and to an apparatus for 5 processing two or more initially decoded audio signals received or replayed from a bitstream, that each have a different number of channels and/or different channel configurations, and that are combined before being presented in a final channel configuration.

10

Background

In the MPEG-4 standard ISO/IEC 14496:2001, in particular in 15 part 3 Audio and in part 1 Systems, several audio objects that can be coded with different MPEG-4 format coding types can together form a composed audio system representing a single soundtrack from the several audio substreams. User interaction, terminal capability, and speaker configuration 20 may be used when determining how to produce a single soundtrack from the component objects. Audio composition means mixing multiple individual audio objects to create a single soundtrack, e.g. a single channel or a single stereo pair. A set of instructions for mixdown is transmitted or transferred in the bitstream. In a receiver the multiple audio 25 objects are decoded separately, but not directly played back to a listener. Instead, the transmitted instructions for mixdown are used to prepare a single soundtrack from the decoded audio objects. This final soundtrack is then played 30 for the listener.

ISO/IEC 14496:2001 is the second version of the MPEG-4 Audio standard, whereas ISO/IEC 14496 is the first version.

In the above MPEG-4 Audio standard nodes for presenting audio are described. Header streams that contain configuration 35 information, which is necessary for decoding the audio substreams are transported via MPEG-4 Systems. In a simple au-

5 dio scene the channel configuration of the audio decoder - for example 5.1 multichannel - can be fed inside the Com-
 5 positor from one node to the following node so that the channel configuration information can reach the presenter,
 10 which is responsible for the correct loudspeaker mapping.
 The presenter represents that final part of the audio chain which is no more under the control of the broadcaster or content provider, e.g. an audio amplifier having volume control and the attached loudspeakers.
 15 'Node' means a processing step or unit used in the above MPEG-4 standard, e.g. an interface carrying out time synchronisation between a decoder and subsequent processing units, or a corresponding interface between the presenter and an upstream processing unit. In general, in ISO/IEC
 20 14496-1:2001 the scene description is represented using a parametric approach. The description consists of an encoded hierarchy or tree of nodes with attributes and other information including event sources and targets. Leaf nodes in this tree correspond to elementary audio-visual data, whereas intermediate nodes group this material to form audio-visual objects, and perform e.g. grouping and transformation on such audio-visual objects (scene description nodes).
 25 Audio decoders either have a predetermined channel configuration by definition, or receive e.g. some configuration information items for setting their channel configuration.

Invention

30 Normally, in an audio processing tree the channel configuration of the audio decoders can be used for the loudspeaker mapping occurring after passing the sound node, see ISO/IEC 14496-3:2001, chapter 1.6.3.4 Channel Configuration. There-
 35 fore, as shown in Fig. 1, an MPEG-4 player implementation passes these information items, that are transmitted within

a received MPEG-4 bitstream, together with the decoder output or outputs through the audio nodes AudioSource and Sound2D to the presenter. The channel configuration data ChannelConfig is to be used by the presenter to make the 5 correct loudspeaker association, especially in case of multi-channel audio (numChan > 1) where the phaseGroup flags in the audio nodes are to be set.

However, when combining or composing audio substreams having different channel assignments, e.g. 5.1 multichannel surround sound and 2.0 stereo, some of the audio nodes (AudioMix, AudioSwitch and AudioFX) defined in the current 10 MPEG-4 standard mentioned above can change the fixed channel assignment that is required for the correct channel representation, i.e. such audio nodes have a channel-variant behaviour 15 leading to conflicts in the channel configuration transmission.

A problem to be solved by the invention is to deal properly with such channel configuration conflicts such that the 20 presenter can replay sound with the correct or the desired channel assignments. This problem is solved by the method disclosed in claim 1. An apparatus that utilises this method is disclosed in claim 3.

25 The invention discloses different but related ways of solving such channel configuration confusion by using channel-variant audio nodes. An additional audio channel configuration node is used, or its functionality is added to the existing audio mixing and/or switching nodes. This additional 30 audio channel configuration node tags the correct channel configuration information items to the decoded audio data streams that pass through the Sound2D node to the presenter.

35 Advantageously, the invention enables the content provider or broadcaster to set the channel configuration in such a way that the presenter at receiver side can produce a cor-

rect channel presentation under all circumstances.

An escape code value in the channel configuration data facilitates correct handling of not yet defined channel combinations even in case signals having different channel configurations even in case signals having different channel configurations are mixed and/or switched together.

The invention can also be used in any other multi-channel application wherein the received channel data are passed through a post processing unit having the inherent ability to interchange the received channels at reproduction.

10

In principle, the inventive method is suited for processing two or more initially decoded audio signals received or replayed from a bitstream, that each have a different number of channels and/or different channel configurations, and that are combined by mixing and/or switching before being presented in a final channel configuration, wherein to each one of said initially decoded audio signals a corresponding specific channel configuration information is attached, and wherein said mixing and/or switching is controlled such that in case of non-matching number of channels and/or types of channel configurations the number and/or configuration of the channels to be output following said mixing and/or following said switching is determined by related specific mixing and/or switching information provided from a content provider or broadcaster, and wherein to the combined data stream to be presented a correspondingly updated channel configuration information is attached.

30 In principle the inventive apparatus includes:

- at least two audio data decoders that decode audio data received or replayed from a bitstream;
- means for processing the audio signals initially decoded by said audio data decoders, wherein at least two of said decoded audio signals each have a different number of channels and/or a different channel configuration, and wherein

said processing includes combination by mixing and/or switching;

- means for presenting the combined audio signals in a final channel configuration, wherein to each one of said initially 5 decoded audio signals a corresponding specific channel configuration information is attached,

- wherein in said processing means said mixing and/or switching is controlled such that in case of non-matching number of channels and/or types of channel configurations 10 the number and/or configuration of the channels to be output following said mixing and/or following said switching is determined by related specific mixing and/or switching information provided from a content provider or broadcaster, and wherein to the combined data stream fed to said presenting 15 means a correspondingly updated channel configuration information is attached.

Advantageous additional embodiments of the invention are disclosed in the respective dependent claims.

20

Drawings

Exemplary embodiments of the invention are described with 25 reference to the accompanying drawings, which show in:

Fig. 1 Transparent channel configuration information flow in a receiver;

Fig. 2 Channel configuration flow conflicts in a receiver;

Fig. 3 Inventive receiver including an additional node 30 AudioChannelConfig.

Exemplary embodiments

35 In Fig. 2 a first decoder 21 provides a decoded '5.1 multichannel' signal via an AudioSource node or interface 24 to

a first input In1 of an AudioMix node or mixing stage 27. A second decoder 22 provides a decoded '2.0 stereo' signal via an AudioSource node or interface 25 to a second input In2 of AudioMix node 27. The AudioMix node 27 represents a multi-channel switch that allows to connect any input channel or channels to any output channel or channels, whereby the effective amplification factors used thereby can have any value between '0'='off' and '1'='on', e.g. '0.5', '0.6' or '0.707'. The output signal from AudioMix node 27 having a '5.1 multichannel' format is fed to a first input of an AudioSwitch node or switcher or mixing stage 28. A third decoder 23 provides a decoded '1 (centre)' signal via an AudioSource node or interface 26 to a second input of AudioSwitch node 28.

The functionality of this AudioSwitch node 28 is similar to that of the AudioMix node 27, except that the 'amplification factors' used therein can have values '0'='off' or '1'='on' only. AudioMix node 27 and Audio switch node 28 are controlled by a control unit or stage 278 that retrieves and/or evaluates from the bitstream received from a content provider or broadcaster e.g. channel configuration data and other data required in the nodes, and feeds these data items to the nodes. Audio switch node 28 produces or evaluates sequences of switching decisions related to the selection of which input channels are to be passed through as which output audio channels. The corresponding whichChoice data field specifies the corresponding channel selections versus time instants. The audio output signal from AudioSwitch node 28 having a '2.0 stereo' format is passed via a Sound2D node or interface 29 to the input of a presenter or reproduction stage 20.

In Fig. 2 two different conflicts are shown. The first conflict occurs in the mix node 27, where a mix of a stereo signal into the surround channels in a 5.1 configuration is shown. The question is, for example, whether the resulting audio output signal should have 5.1 channels, or the 5.1

surround channels should become 2.0 stereo format channels. In case of selecting a 5.1 output format the straight-forward solution would be to assign input signal L2 to the first output channel 1ch and input signal R2 to the second 5 output channel 2ch. However, there are many other possibilities. The content provider or broadcaster could desire to assign input signal L2 to output channel 4ch and input signal R2 to output channel 5ch instead. However, the current version of the above MPEG-4 standard does not allow such 10 feature.

The second conflict occurs in the sequence of whichChoice data field updates in the AudioSwitch node 28. Within this sequence, channels out of the AudioMix node 27 output and the single channel output from AudioSource node 26 are sequentially selected at specified time instants. The time instants in the whichChoice data field can be defined by e.g. every succeeding frame or group of frames, every predetermined time period (for instance 5 minutes), each time the content provider or broadcaster has preset or commanded, or 15 upon each mouse click of a user. In the example given in Fig. 2, at a first time instant input signal C1 is connected to output channel 1ch and input signal M is connected to output channel 2ch. At a second time instant input signal L1 is connected to output channel 1ch and input signal R1 is 20 connected to output channel 2ch. At a third time instant input signal LS1 is connected to output channel 1ch and input signal RS1 is connected to output channel 2ch. Within this sequence, channels out of the AudioMix node 27 output and the single channel output from AudioSource node 26 are sequentially selected. However, because of the contradictory 25 input information in node 28, no correct output channel configuration can be determined automatically based on the current version of the above MPEG-4 standard.

35 Based on the assumption that the content provider or broadcaster is to solve such conflicts, three inventive solutions

are feasible that are explained in connection with Fig. 3. A first decoder 21 provides a decoded '5.1 multichannel' signal via an AudioSource node or interface 24 to a first input of an AudioMix node or mixing stage 27. A second decoder 22 provides a decoded '2.0 stereo' signal via an AudioSource node or interface 25 to a second input of AudioMix node 27. The output signal from AudioMix node 27 having a '5.1 multichannel' format is fed to a first input of an AudioSwitch node or switcher or mixing stage 28. A third decoder 23 provides a decoded '1 (centre)' signal via an AudioSource node or interface 26 to a second input of AudioSwitch node 28. The decoders may each include at the input an internal or external decoding buffer. The output signal from AudioSwitch node 28 having a '2.0 stereo' format is passed via a Sound2D node or interface 29 to the input of a presenter or reproduction stage 20.

AudioMix node 27 and Audio switch node 28 are controlled by a control unit or stage 278 that retrieves and/or evaluates from the bitstream received from a content provider or broadcaster e.g. channel configuration data and other data required in the nodes, and feeds these data items to the nodes.

A new audio node, called AudioChannelConfig node 30 is introduced between AudioSwitch node 28 and Sound2D node 29. This node has the following properties or function:

AudioChannelConfig{

exposedField	SFInt32	numChannel	0
exposedField	MFInt32	phaseGroup	0
exposedField	MFInt32	channelConfig	0
30 exposedField	MFFloat	channelLocation	0,0
exposedField	MFFloat	channelDirection	0,0
exposedField	MFInt32	polarityPattern	1

},

expressed in the MPEG-4 notation. SFInt32, MFInt32 and MFFloat are single field (SF, containing a single value) and multiple field (MF, containing a multiple values and the

quantity of values) data types that are defined in ISO/IEC 14772-1:1998, subclause 5.2. 'Int32' means an integer number and 'Float' a floating point number. 'exposedField' denotes a data field the content of which can be changed by the content provider or broadcaster per audio scene.

The phaseGroup (*specifies phase relationships in the node output, i.e. specifies whether or not there are important phase relationships between multiple audio channels*) and the numChannel (*number of channels in the node output*) fields 10 are re-defined by the content provider due to the functional correlation with the channelConfig field or parameters.

The channelConfig field and the below channel configuration association table can be defined using a set of pre-defined index values, thereby using values from the ISO/IEC 14496-15 3:2001 audio part standard, chapter 1.6.3.4. According to the invention, it is extended using some values of chapter 0.2.3.2 of the MPEG-2 audio standard ISO/IEC 13818-3:

index value	No. of channels	audio syntactic elements, listed in order received	Channel to speaker mapping
0	unspecified	unspecified	channelConfiguration from child node is passed through
1	-	Escape sequence	The channelLocation, channelDirection and polarityPattern fields are valid
2	1	single_channel_element	centre front speaker
3	2	channel_pair_element	left, right front speakers
4	3	single_channel_element, channel_pair_element	centre front speaker, left, right front speakers
5	4	single_channel_element, channel_pair_element, single_channel_element	centre front speaker, left, right centre front speakers, rear surround speakers

6	5	single_channel_element, channel_pair_element, channel_pair_element	centre front speaker, left, right front speakers, left surround, right surround rear speakers
7	5+1	single_channel_element, channel_pair_element, channel_pair_element, lfe_element	centre front speaker, left, right front speakers, left surround, right surround rear speakers, front low frequency ef- fects speaker
8	7+1	single_channel_element, channel_pair_element, channel_pair_element, channel_pair_element, lfe_element	centre front speaker, left, right centre front speakers, left, right outside front speakers, left surround, right surround rear speakers, front low frequency ef- fects speaker
9	2/2	MPEG-2 L, R, LS, RS	left, right front speakers, left surround, right surround rear speakers
10	2/1	MPEG-2 L, R, S	left, right front speakers, rear surround speaker
....			

Table 1: Channel configuration association

Advantageously, an escape value '1' is defined in this table having e.g. index '1' in the table. If this value occurs, 5 the desired channel configuration is not listed in the table and therefore the values in the channelLocation, channelDirection and polarityPattern fields are to be used for assigning the desired channels and their properties. If the channelConfig index is an index defined in the table, the 10 channelLocation, channelDirection and polarityPattern fields are vectors of the length zero.

In the channelLocation and channelDirection fields a 3D-float vector array can be defined, whereby the first 3 float

values (three-dimensional vector) are associated with the first channel, the next 3 float values are associated with the second channel, and so on.

5 The values are defined as x,y,z values (*right handed coordinate system as used in ISO/IEC 14772-1 (VRML 97)*). The channelLocation values describe the direction and the absolute distance in meter (the absolute distance has been used because simply the user can generate a normalised vector, as usually used in channel configuration). The channelDirection 10 is a unit vector with the same coordinate system. E.g. channelLocation [0, 0, -1] relative to the listening sweet spot means centre speaker in one-meter distance. Three other examples are given in the three lines of table 2:

channelLocation			channelDirection			Location
x	y	z	x	y	z	
0	0	-1	0	0	1	center front speaker
$k \cdot \sin(30^\circ)$	0	$k \cdot -\cos(60^\circ)$	$-\sin(30^\circ)$	0	$\cos(60^\circ)$	right front speaker
$k \cdot \sin(45^\circ)$	$k \cdot \sin(45^\circ)$	$k \cdot -\cos(45^\circ)$	$\sin(45^\circ)$	$-\sin(45^\circ)$	$\cos(45^\circ)$	Ambisonic Cube (LFU) Left Front Up

15 Table 2: Examples for channelLocation and channelDirection

The polarityPattern is an integer vector where the values are restricted to the values given in table 3. This is useful for example in case of Dolby ProLogic sound where the 20 front channels have monopole pattern and the surround channel have dipole characteristic.

The polarityPattern can have values according to table 2.

Value	Characteristics
0	Monopole
1	Dipole
3	Cardioide
4	Headphone
...	...

Table 1: polarityPattern association

In an alternative embodiment of the invention, the additional AudioChannelConfig node 30 is not inserted. Instead, the functionality of this node is added to nodes of the type AudioMix 27, AudioSwitch 28 and AudioFX (not depicted).

5

In an further alternative embodiment of the invention, the above values of the phaseGroup fields are additionally defined for the corresponding existing nodes AudioMix, AudioSwitch and AudioFX in the first version ISO/IEC 14496 of the MPEG-4 standard. This is a partial solution whereby the values for the phase groups are taken from above table 1 except the escape sequence. Higher values are reserved for private or future use. For example, channels having the phaseGroup 2 are identified as left/right front speakers.

10
15